

Project 2 Experiment Write-up

Experimental design:

Input file: see lorum_ispum.txt

Regexps:

```
#1. /^Loremipsumdolorsitametconsecteturadipiscingelit$/
#2. /^Integertempusdiamnonorciplaceratquisaliquamliberoporta$/
#3. /^Integerposuereaccumsanbibendum$/
#4.
/^Invelquamnequevehicularutrumlibero|Vestibulumliberonibhvulputateseda
dipiscingeumolestiequislacusNullaquissapieneratidpretiumerat$/
#5.
/^Nullamaturpismetusnecblanditlacus|Nullamenimodiocondimentumetmollisv
elconvallisnonpurusSedpretiumviverraauctor$/
#6. /^Nuncornarevehiculaestinfaucibus$/
#7.
/^CrasornareportamiidcommodoMorbiimperdietdolorutenimultricesfeugiat$/
```

The regular expressions are run through the text file lorum_ispum. If it accepts as a dfa, the program returns the time it took to run. Most lines of string test the concat function of regexp. #4 and #5 test the union function. I was unsure how to test the * function though.

Computer equipment:

Asus EeePC:

OS: Windows 7 starter edition

1.6 GHz Intel Atom Processor

1 GB RAM

160 GB Harddrive

Summary: See results.txt

The difference between the DFA and NFA output varies but those compared to the Ruby regexp implementation takes much longer. Sometimes the DFA will spit out the correct output before the NFA and other times, the NFA outputs faster. Overall, the NFA works faster than the DFA. Mostly on the longer strings, but I can't see too much of a big difference when I look at each case.